**ЗАДАНИЕ**

**на курсовую работу**

1. Тема работы: Контроль и учет технического состояния магистральных трубопроводов, транспортирующих огнеопасные продукты.
2. Техническое задание: Разработать и создать базу данных и приложение для взаимодействия с базой контроля и учета технического состояния магистральных трубопроводов, транспортирующих огнеопасные вещества.
3. Объём и содержание проекта (графических работ 2 листа). База данных, и программное средство, состоящее из программы взаимодействующей с базой. Пояснительная записка, содержащая: введение, аналитическую часть, описывающую основные концепции разработки баз данных, конструкторскую и технологическую часть описывающих архитектуру программного средства и работу с ним, заключение и список литературы.

**Оглавление**

# ВВЕДЕНИЕ

# 1 АНАЛИТИЧЕСКАЯ ЧАСТЬ

# 1.1 Постановка задачи

# 1.2 Сущности, отношения и их свойства

# 1.3 Условия целостности

# 1.4 Нормальные формы таблиц

# 1.5 Бизнес правила

# 1.6 Требования к БД

# 2 КОНСТРУКТОРСКАЯ ЧАСТЬ

# 2.1 Структура системы

# 2.2 Проектирование базы данных

# 2.3 Хранимые процедуры

# 2.4 Запросы

# 2.5 Выбор модели базы данных

# 2.6 Организация взаимодействия с пользователем

# 3 ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ

# 3.1 Выбор средства разработки

# 3.2 Выбор СУБД

# 3.3 Описание системы

# 3.4 Синхронизация при работе в сети

# 3.5 Технические характеристики программы

# 4 ИССЛЕДОВАТЕЛЬСКАЯ ЧАСТЬ

# ЗАКЛЮЧЕНИЕ

# СПИСОК ЛИТЕРАТУРЫ

# Приложение 1

# ВВЕДЕНИЕ

Магистральный трубопровод – разветвленная система транспортировки нефтепродуктов, газа и других веществ промышленного использования. В случае транспортировки огнеопасных продуктов – это объект представляющий повышенную опасность для окружающей среды, за которым необходим постоянный технический контроль. Для наиболее точного контроля за техническим состоянием магистральных трубопроводов и быстрого устранения возникших неисправностей необходима автоматизированная система, которая содержала бы всю информацию о магистральных трубопроводах и в случае необходимости ей могли бы воспользоваться сотрудники подразделений, отвечающих за эксплуатацию данных объектов.

Использование баз данных для сведения всей информации о данных объектах выглядит наиболее целесообразным решением проблемы контроля за состоянием магистральных трубопроводов по следующим причинам:

* удобный и компактный способ хранения данных;
* возможность многопользовательского доступа упрощающего формирование базы данных;
* возможность удаленного доступа;
* возможность построения систем поисков, сортировок и, следовательно, увеличение скорости работы с большими массивами документов.

Крупные промышленные компании (такие как Лукойл, Газпром, Юкос), представляющие собой удаленные от головного центра региональные предприятия особо остро нуждаются в создании таких систем.

Такая база данных позволит получать актуальную информацию о состоянии трубопровода на любом участке его протяженности за счет интерактивной модификации хранящихся данных и возможности многопользовательского удаленного управления.

При правильном выборе СУБД и корректном создании базы данных исключается возможность злоумышленного изменения параметров объекта, которые могут быть критичны для оценки ситуации и всей работы предприятия.

В целом создание и развитие такой системы будет полностью отвечать постоянно растущим требованиям автоматизации производства.

# АНАЛИТИЧЕСКАЯ ЧАСТЬ

## Постановка задачи

Целью данного курсового проекта является разработка и создание базы данных контроля и учета технического состояния магистральных трубопроводов, транспортирующих огнеопасные продукты. Среди основных задач работы можно выделить три основных:

* Детальная разработка структуры базы данных и ее основных связей;
* Анализ существующих СУБД. Выбор одной СУБД, отвечающей требованиям надежности и имеющие необходимы для создания такой системы инструментарий;
* Создание базы данных.

Разрабатываемая система должна предоставлять средства для сбора информации от подразделений, отвечающих за техническое состояние трубопроводов. Собранные данные должны храниться в базе данных. Необходимо разработать пользовательский интерфейс к базе данных. Интерфейс должен предоставлять следующие возможности:

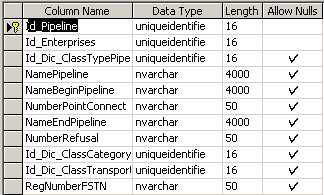
* доступ к базе через локальную вычислительную сеть;
* редактирование данных в базе;
* возможность проведения поисков и сортировок по атрибутам.

В данном курсовом проекте поставлена задача, разработать автоматизированную систему контроля техническим состоянием трубопроводов транспортирующих огнеопасные вещества.

## Сущности, отношения и их свойства

При описании предметной области с точки зрения концептуальной модели, прежде всего, следует определить сущности, принадлежащие этой области, и связи между ними. Под сущностью, в таком подходе, понимается то, о чем должна накапливаться и обрабатываться информация. Например, при разработке схемы функционирования факультета сущностями могут выступать студенты факультета, преподаватели, читаемые предметы, методический и научно-исследовательский материал, разрабатываемый факультетом, семинары и конференции, проводимые на данном факультете и так далее. Каждая сущность характеризуется с помощью ограниченного набора свойств и связей с другими сущностями. Группа сущностей, характеризующаяся одним и тем же набором свойств, образует набор сущностей. Так, например, список студентов образует набор сущностей, который мы назовем СТУДЕНТ, и он будет характеризоваться следующими свойствами: фамилия, имя и отчество; номер студенческого билета; группа; место жительства; год поступления; наличие или отсутствие стипендии и тому подобное. Свойства набора сущностей называют атрибутами, а множество допустимых значений атрибутов называют доменом. С точки зрения датологической модели при описании атрибутов каждого из набора сущностей, следует указать не только имя атрибута, но и тип данных, описывающих данный атрибут. Тип данных, используемых при описании атрибута, зависит от того смысла, который вкладывается в этот атрибут при проектировании модели объектной области. Например, если в наборе объектов СТУДЕНТ атрибут «стипендия» характеризует только ее наличие или отсутствие, то есть домен этого атрибута состоит всего лишь из двух значений, то для его описания следует использовать логический тип. Если же этот атрибут описывает истинное значение стипендии, то тогда его значение должно быть числовым или денежным. Если же этот атрибут характеризует тип стипендии, например, обычная, повышенная, именная и так далее, то тип данных, отвечающих такому атрибуту, следует задать литерным.

В качестве примера, рассмотрим набор объектов, характеризующий сотрудников некоторой фабрики. В качестве атрибутов можно указать следующее:



Следует отметить, что в наборе сущностей должна обеспечиваться возможность выделить конкретную сущность из набора. Для однозначной идентификации конкретной сущности вводится понятие ключа. Ключом может служить или конкретный атрибут (простой ключ) или некоторая совокупность атрибутов (сложный или составной ключ).

При проектировании концептуальной модели первым важным вопросом является выбор набора сущностей, с помощью которого полностью охватывается интересующая нас часть предметной области. Второй вопрос – это выбор атрибутов, подходящих для описания этих наборов сущностей. Третий важный вопрос – это установление связей между наборами сущностей и их описание.

Связь между наборами объектов может быть трех типов. Первый тип – связь один к одному (обозначение 1:1), когда между записями двух наборов сущностей устанавливается связь, характеризующаяся взаимно однозначным соответствием между сущностями, входящими в каждый из наборов. Например, если один из наборов сущностей – это номера проданный на данный рейс билетов, а другой – это список пассажиров, то связь между ними будет один к одному. При нарушении этого принципа должен выдаваться сигнал ошибки, так как на одно и то же место будет продано несколько билетов. Второй тип связи – это один ко многим (1:М), или обратный вариант – многие к одному (М:1). Например, если один набор сущностей это клиенты некоторого банка, а другой – счета банка, то если у клиента в банке допускается несколько счетов, то будет установлена связь один ко многим. В случае, когда первичным рассматривается счет, то связь будет трактоваться как многие к одному. Третий вид связи – это многие ко многим (M:N), когда нескольким записям одного набора сущностей соответствует несколько записей другого набора. В качестве примера можно рассмотреть список студентов некоторого факультета и список предметов, читаемых на этом факультете. Связь между этими наборами сущностей будет как раз определяться как многие ко многим, причем она усложнится, если для студентов на факультете допускается некоторый выбор изучаемых предметов.

В реляционной модели данных сущность набор сущностей интерпретируется в виде таблиц, называемых отношениями или реляциями. В отношениях столбцы представляют собой атрибуты, и им присваиваются имена, по которым затем происходит обращение. Кортеж, соответствующий данной схеме отношения, представляет собой множество пар {имя атрибута, значение}, которое содержит одно вхождение каждого имени атрибута, принадлежащего данному отношению.

## Условия целостности

Одним из условий нормального функционирования базы данных является понятие целостности данных, где модно выделить два аспекта.

* категорийная целостность;
* ссылочная целостностью

Категорийная целостность связана с тем, что значения атрибутов, являющихся ключом отношения, должны быть уникальными и не могут быть неопределенными. Поэтому строка, являющаяся кортежем не может быть занесена в отношение до тех пор, пока не будут определены все атрибуты ее ключа. что касается ссылочной целостности, то если в одном отношении удаляется запись, то в отношениях, связанных с ним не должно быть кортежей, имеющих значение внешнего ключа удаляемой записи.

При использовании SQL Server 2000 для построения базы данных отношения трансформируются в таблицы, а при установлении связей с помощью схемы данных необходимо позаботиться об обеспечении целостности и каскадном обновлении и удалении данных.

## Нормальные формы таблиц

Первая нормальная форма.

Таблица представлена в первой нормальной форме (1НФ) тогда и только тогда, когда все ее столбцы содержат только неделимые (в том смысле, что их дальнейшее разложение невозможно) значения и в ней отсутствуют повторяющиеся группы (столбцов) в пределах одной строки.

Первое правило реляционной модели состоит в том, что ни один столбец не должен содержать два или более значений, т.е. реляционная модель базы данных запрещает повторяющиеся поля.

Ни одну из реляционных СУБД не удовлетворяет только 1НФ, поскольку в этом случае требуется определять большое число полей, многие из которых остаются в основном пустыми. Альтернативный подход, при котором записи могут иметь переменную длину, практически не применяется из-за больших затрат ресурсов на реализацию алгоритма, определяющего, где кончается одна запись и начинается следующая.

Метод, который можно использовать для приведения таблицы к первой нормальной форме, состоит в том, чтобы разбить ее данные на две связанные таблицы

Если таблица находится в 1НФ, то отображение ее данных и построение к ней запросов не составляет никакой сложности. Однако это правило еще не препятствует дублированию данных в строках. Следствием будет ввод в таблицу большого количества данных, хранить которые нет никакой необходимости. Избыточные данные могут послужить причиной проблем целостности и снижения эффективности при внесении изменений, поэтому подобных решений при проектировании баз данных необходимо избегать.

Вторая нормальная форма.

Для определения второй нормальной формы необходимо ввести концепцию функциональной зависимости. Это зависимость, связывающая атрибуты в одной таблице с единственным значением в другой таблице. Функциональную зависимость для таблиц А и В принято обозначать как А--В. Это понятие подводит "на один шаг" к родственной концепции объединения таблиц в отношения типа 1:1 или 1:М.

Таблица представлена во второй нормальной форме (2НФ) тогда и только тогда, когда она представлена в 1НФ каждый не ключевой атрибут полностью определяется первичным ключом. Атрибут полностью определяется первичным ключом, если он находится в правой части выражения, описывающего функциональную зависимость, а левую часть этого выражения представляет первичный ключ или какое-либо выражение, которое может быть вычислено на его основе с использованием транзитивных свойств функциональной зависимости.

Избыточность 1НФ может послужить причиной возникновения проблем и при удалении или добавлении. Аномалия вставки может проявиться в том случае, когда при добавлении в таблицу с первичным ключом какой-либо записи вы вынуждены поместить в поле первичного ключа либо пустое, либо уже существующее значение. Подобное действие нарушает основные требования реляционной теории и потребует от вас (как минимум) заново сформировать поле первичного ключа этой таблицы и все связанные с ним ключевые поля в других таблицах, что вызовет значительные потери в производительности. Аналогично, аномалия удаления проявляется в тех случаях, когда у вас возникает необходимость удалить запись и провести реорганизацию ваших таблиц. Если впоследствии потребуется восстановить удаленную запись, то вы вынуждены будете вновь подвергнуть обработке весь набор таблиц. Цель разработки 2НФ состоит в устранении этих проблем.

Третья нормальная форма.

Если A является элементом кандидатного ключа, то будем называть A первичным атрибутом. Под тривиальной функциональной зависимостью здесь понимается зависимость типа xA -> A. Эта зависимость тривиальна, поскольку Х не принимает участия в идентификации первичного атрибута, лишь подразумевается в его определении.

Это определение - всего лишь оригинальный способ выразить необходимость представления системы связанных таблиц в таком виде, чтобы атрибуты каждой таблицы непосредственно определялись либо суперключом, либо кандидатным ключом этой таблицы. В предыдущем примере реализация определения ЗНФ состоит в выделении таблицы Контролирующие лица.

Для большинства существующих СУБД и инструментов CASE-технологии необходимо представить проект вашей базы данных в ЗНФ, так как этого вполне достаточно практически для всех обычных приложений. При разработке исключительно больших систем на сверхбыстродействующих компьютерах, когда необходимо обеспечить максимальное сокращение объемов хранимых данных, желательно провести дальнейшую нормализацию.

Таблица Х представлена в нормальной форме Бойса-Кодда (БКНФ), если в каждой нетривиальной функциональной зависимости В -> А, В является суперключом.

Уместно сделать несколько замечаний о недостатках, присущих даже таблицам, представленным в ЗНФ. Существуют варианты, когда имеет смысл разделить таблицу на более мелкие таблицы, если часть представленных в ней данных непостоянна и часто обновляется, а остальные данные пассивны и изменяются в редких случаях. Также есть смысл объединить таблицы, когда необходимо обеспечить высокую скорость реакции на запрос. Можно даже пойти на дублирование данных в таблицах, если это позволит снизить затраты на обработку запросов, хотя формально не следовало бы этого делать. Иногда приходится отступить от принципа полной нормализации данных проекта. Это может быть вызвано следующими причинами:

* временем выполнения запросов;
* временем проведения обновлений;
* общим необходимым объемом хранилища данных;
* аномалиями удаления, которые могут вызвать потерю целостности данных.

И ЗНФ и БКНФ являются теоретическими конструкциями, в то время как большинство разработчиков баз данных работают в реальном мире.

Четвертая и пятая нормальные формы.

Прежде чем закончить рассмотрение правил Кодда, вам будет предложен краткий обзор двух последних правил реляционных баз данных. Эти два правила предназначены для устранения еще двух аномалий, называемых многозначная зависимость и объединяющая зависимость.

Многозначная зависимость определяется следующим образом.

В таблице Х существует многозначная зависимость А -> В, если в этой таблице можно обнаружить ситуации, в которых пара строк содержит дублирующееся значение А и одновременно существуют другие пары строк, полученные путем перестановки значений В, присутствующих в первой паре.

Прежде всего, для существования многозначной зависимости требуется существование пар строк. А и В могут быть как отдельными атрибутами, так и объединением некоторого набора атрибутов. Тривиальная многозначная зависимость для А -> В существует в случае, когда В является подмножеством А или А объединяет B = xS (более крупная таблица содержит исходную таблицу). Существование многозначной зависимости порождает аномалию обновления. Четвертая нормальная форма устраняет нетривиальную многозначную зависимость в таблице посредством создания меньших таблиц. Процесс нормализации представляет собой создание как можно большего числа все более мелких таблиц в целях сокращения избыточности данных.

Определим четвертую нормальную форму.

Таблица Х представлена в 4НФ тогда и только тогда, когда она представлена в БКНФ и для любой многозначной зависимости А -> В в этой таблице можно сказать, что эта зависимость либо является тривиальной, либо А является суперключом таблицы X.

И последнее. Рассмотрим пятую нормальную форму.

Пятая нормальная форма достигается в том случае, когда таблица не может далее разбиваться на более мелкие таблицы посредством операции проектирования.

Под операцией проектирования понимается не вызывающая потерь данных декомпозиция, при которой таблица разбивается на части таким образом, что остается возможность объединения полученных меньших таблиц для получения в результате исходной таблицы.

## Бизнес правила

1. Факты

* При создании трубопровода создается участок трубопровода.

1. Ограничения

* Пользователь не может удалить участок трубопровода, если их количество не превышает одного.
* Пользователь не может сохранить изменения в базе пока не введет все необходимые данные.

1. Активаторы

* Если срок эксплуатации участка трубопровода истек, то необходимо уведомить пользователя об этом.
* Если в базе хранятся не все необходимы данные о трубопроводе, то необходимо уведомить пользователя об этом.

1. Вывод

* Если удаляется трубопровода, то удаляются все его участки, комментарии к нему и обследования с целью продления срока службы.
* Если удаляется участок трубопровода, то удаляются все его испытания, ревизии, отказы, ремонты, диагностики.

1. Вычисления

* Общая длина трубопровода вычисляется как сумма длин участков трубопровода.
* При вводе даты последнего обследования и выбора периодичности обследования, вычисляется дата следующего обследования.

## Требования к БД

* доступ через локальную вычислительную сеть;
* вся информация должна представляться в определенном виде в соответствии с разработанной структурой;
* в БД должен предусмотрен поиск;
* в качестве минимальных входных данных должны рассматриваться:
  + - вид трубопровода;
    - тип трубопровода;
    - наименование трубопровода.
* предупреждать пользователя если он не ввел все данные о трубопроводе или участке трубопровода;
* выдавать сообщения пользователю о наиболее важных изменениях в состоянии трубопровода и не введенных наиболее важных данных.

# КОНСТРУКТОРСКАЯ ЧАСТЬ

## 

## Структура системы

Для реализации поставленной задачи необходимо разработать три подсистемы:

* подсистема хранения данных;
* подсистема доступа к базе данных;

Подсистема хранения данных представляет собой реляционную базу данных.

Подсистема доступа к базе данных должна предоставлять пользовательские функции для работы с информацией о магистральных трубопроводах.

## Проектирование базы данных

Для реализации поставленной задачи необходимо создать таблицы:

* «Трубопроводы»,
* «Участки трубопроводов»,
* «Примечание»,
* «Обследование с целью продления срока службы»,
* «Ремонт на участках»,
* «Ревизия участков трубопроводов»,
* «Диагностика участков трубопроводов»,
* «Испытание участков трубопроводов»,
* «Отказы на участках».

«Трубопроводы». В этой таблице хранятся основные данные.

* «Вид». Значение этого атрибута является вид трубопровода. Вид трубопровода выбирается пользователем из справочника «Виды трубопровода».
* «Наименование трубопровода». Значением этого атрибута является название трубопровода, задаваемое пользователем.
* «Наименование начала линии(трубы)». Значением этого атрибута является название начала линии, задаваемой пользователем.
* «№ точки подключения». Значением этого атрибута является номер точки подключения, задаваемой пользователем.
* «Наименование конца линии(трубы)». Значением этого атрибута является название конца линии, задаваемой пользователем.
* «Количество отказов». Значением этого атрибута является числовое значение, соответствующее количеству отказов трубопровода.
* «Категория трубопровода». Значением этого атрибута является категория трубопровода, выбираемая пользователем из справочника «Категория трубопровода».
* «Транспортируемый продукт». Значением этого атрибута является транспортируемый продукт, выбираемый пользователем из справочника «Транспортируемые продукты».
* «Регистрационный № ФСТН». Значением этого атрибута является номер регистрации в ФСТН, который выдается при регистрации объекта.
* «Дата регистрации в ФСТН». Значением этого атрибута является дата регистрации в ФСТН.
* «Регистрационный № СТН(ОТН)». Значением этого атрибута является номер регистрации в СТН, который выдается при регистрации объекта.
* «Дата регистрации в СТН(ОТН)». Значением этого атрибута является дата регистрации в СТН.
* «Регистрационный № СПК». Значением этого атрибута является номер регистрации в СТН, который выдается при регистрации объекта
* «Дата регистрации в СПК». Значением этого атрибута является дата регистрации в СПК.
* «Дата снятия с регистрации в ФСТН». Значением этого атрибута является дата снятия с регистрации в ФСТН.
* «Дата снятия с регистрации в СТН(ОТН)». Значением этого атрибута является дата снятия с регистрации в СТН.
* «Дата снятия с регистрации в СПК». Значением этого атрибута является дата снятия регистрации в СПК.
* «Организация местоположения». Значением этого атрибута является полный список организаций, которым принадлежит трубопровод.
* «Месторождение». Значением этого атрибута является наименования месторождение, выбираемое пользователем из справочника «Месторождения».
* «Место установки». Значением этого атрибута является наименование предприятия, где установлен трубопровод.
* «Отношение к ОПО». Значением этого атрибута является наименование опасных промышленных объектов, который выбирается пользователем.
* «Владелец». Значением этого атрибута является наименование владельца предприятия которому принадлежит трубопровод.
* «Дополнительная информация». Значением этого атрибута является дополнительные сведения о трубопроводе, вводимые пользователем.

«Участки трубопроводов». В этой таблице хранятся данные, которые содержат информацию об участках трубопроводов.

* «Трубопровод». Значением этого атрибута является указатель на трубопровод к которому принадлежит примечание.
* «Наименование участка». Значением этого атрибута является название участка трубопровода, задаваемое пользователем.
* «Дата врезки». Значением этого атрибута является дата, когда была осуществлена врезка трубопровода.
* «Протяженность (м)». Значением этого атрибута является протяженность участка трубопровода.
* «Марка стали, ГОСТ или ТУ». Значением этого атрибута является марка стали из которой сделан трубопровод.
* «Наружный диаметр (мм)». Значением этого атрибута является наружный диаметр трубопровода.
* «Рабочее давление, кгс/кв.см(Мпа)». Значением этого атрибута является рабочее давление трубопровода.
* «Рабочая температура стенки, град. С». Значениме этого атрибута является рабочая температура стенки участка трубопровода.
* «Тип изоляции». Значением этого атрибута является тип изоляции, выбираемый пользователем из справочника «Тип изоляции».
* «Тип попутного обогрева». Значением этого атрибута является тип попутного обогрева, выбираемый пользователем из справочника «Тип попутного обогрева».
* «Марка ингибитора». Значением этого атрибута является марка ингибитора, выбираемая пользователем из справочника «Марка ингибитора».
* «Номинальная толщина (мм)». Значением этого атрибута является номинальная толщина участка трубопровода.
* «Фактическая толщина стенок (мм)». Значением этого атрибута является фактическая толщина стенок, которая вычисляется при проведении экспертизы.
* «Скорость коррозии, мм/год». Значением этого атрибута является скорость коррозии трубопровода.
* «Номер проекта». Значением этого атрибута является номер проекта.
* «Наименование проектной организации». Значением этого атрибута является наименование проектной организации, которая выбирается пользователем из справочника «Проектные организации»
* «Наименование строительно-монтажной организации». Значением этого атрибута является наименование строительно-монтажной организации, которая выбирается пользователем из справочника «Строительно-монтажные организации».
* «Дата окончания строительства». Значением этого атрибута является дата окончания строительства участка трубопровода.
* «Дата ввода в эксплуатацию». Значением этого атрибута является дата ввода в эксплуатацию участка трубопровода.
* «Дата окончания службы». Значением этого атрибута является дата окончания срока службы участка трубопровода.
* «Выявленные отклонения от технических параметров». Значением этого атрибута является выявленные отклонения от технических параметров в ходе эксплуатации.
* «Дата вывода из эксплуатации». Значением этого атрибута является дата вывода участка трубопровода из эксплуатации.
* «Причина вывода из эксплуатации». Значением этого атрибута является информации содержащая причины вывода участка трубопровода из эксплуатации.

«Примечание». В этой таблице хранятся данные, которые могут понадобиться для правильного и быстрого анализа ситуации.

* «Трубопровод». Значением этого атрибута является указатель на трубопровод к которому принадлежит примечание.
* «Дата примечания». Значением данного атрибута является дата, когда примечание было добавлено.
* «Примечание». Значением этого атрибута является дополнительные сведения о трубопроводе, вводимые пользователем.

«Обследование с целью продления срока службы». В этой таблице хранятся данные, содержащие информацию об обследовании проводившегося с целью продления срока службы трубопровода.

* «Трубопровод». Значением этого атрибута является указатель на трубопровод к которому принадлежит обследование.
* «Дата обследования с целью продления службы». Значением этого атрибута является дата проведения обследование для продления срока службы трубопровода.
* «Наименование организации, проводившей обследование». Значением этого атрибута является наименование организации проводившей обследование.
* «Ф.И.О. эксперта». Значением этого атрибута является Ф.И.О. эксперта, который проводил обследование трубопровода.
* «Выявленные отклонения от технических параметров». Значением этого атрибута является выявленные экспертом отклонения от нормы.
* «Дата завершения эксплуатации». Значением этого атрибута является дата завершения эксплуатации трубопровода.
* «Заключение». Значением этого атрибута является заключение составленное экспертом в результате обследования трубопровода.
* «Номер заключения». Значением этого атрибута является номер заключения.
* «Регистрационный номер заключения в ФСТН». Значением этого атрибута является номер заключения в ФСТН, которое регистрируется после обследования трубопровода и составления экспертом заключения.

«Ревизия участков». В этой таблице хранятся данные, содержащие всю информацию о проводившихся ревизиях на участке трубопровода.

* «Участок трубопровода». Значением этого атрибута является указатель на участок трубопровода к которому относится ревизия.
* «Дата обследования». Значением этого атрибута является дата проведения ревизии на участке трубопровода.
* «Наименование организации, проводившей обследование». Значением этого атрибута является наименование организации проводившей ревизию участка трубопровода.
* «Ф.И.О. эксперта». Значением этого атрибута является Ф.И.О. эксперта, который проводил ревизию участка трубопровода.
* «Дата следующего обследования». Значением этого атрибута является дата следующего проведения ревизии.
* «Периодичность». Значением этого атрибута является периодичность проведения ревизий на участке трубопровода.
* «Дата заключения». Значением этого атрибута является дата заключения, когда было сделано заключение на основе ревизии.
* «Номер заключения». Значением этого атрибута является номер заключения.
* «Заключение». Значением этого атрибута является заключение которое было сделано экспертом в ходе проведения ревизии.

«Испытания участков». В этой таблице хранятся данные, содержащие всю информацию о проводившихся испытаниях на участке трубопровода.

* «Участок трубопровода». Значением этого атрибута является указатель на участок трубопровода к которому относится испытание.
* «Дата обследования». Значением этого атрибута является дата проведения испытания на участке трубопровода.
* «Наименование организации, проводившей обследование». Значением этого атрибута является наименование организации проводившей испытание участка трубопровода.
* «Ф.И.О. эксперта». Значением этого атрибута является Ф.И.О. эксперта, который проводил испытание участка трубопровода.
* «Дата следующего обследования». Значением этого атрибута является дата следующего проведения испытания.
* «Периодичность». Значением этого атрибута является периодичность проведения испытаний на участке трубопровода.
* «Дата заключения». Значением этого атрибута является дата заключения, когда было сделано заключение на основе испытания.
* «Номер заключения». Значением этого атрибута является номер заключения.
* «Заключение». Значением этого атрибута является заключение которое было сделано экспертом в ходе проведения испытания.

«Диагностика участков». В этой таблице хранятся данные, содержащие всю информацию о проводившихся диагностиках на участке трубопровода.

* «Участок трубопровода». Значением этого атрибута является указатель на участок трубопровода к которому относится диагностика.
* «Дата обследования». Значением этого атрибута является дата проведения диагностики на участке трубопровода.
* «Наименование организации, проводившей обследование». Значением этого атрибута является наименование организации проводившей диагностику участка трубопровода.
* «Ф.И.О. эксперта». Значением этого атрибута является Ф.И.О. эксперта, который проводил диагностику участка трубопровода.
* «Дата следующего обследования». Значением этого атрибута является дата следующего проведения диагностики.
* «Периодичность». Значением этого атрибута является периодичность проведения диагностик на участке трубопровода.
* «Дата заключения». Значением этого атрибута является дата заключения, когда было сделано заключение на основе диагностики.
* «Номер заключения». Значением этого атрибута является номер заключения.
* «Заключение». Значением этого атрибута является заключение которое было сделано экспертом в ходе проведения диагностики.

«Ремонт участков». В этой таблице хранятся данные, содержащие всю информацию о проводившихся ремонтах на участке трубопровода.

* «Участок трубопровода». Значением этого атрибута является указатель на участок трубопровода к которому относится ремонт.
* «Дата поломки». Значением этого атрибута является дата поломки участка трубопровода.
* «Дата начала ремонта». Значением этого атрибута является дата начала ремонта участка трубопровода.
* «Заключение о поломке». Значением этого атрибута является заключение о поломке участка трубопровода.
* «Дата конца ремонта». Значением этого атрибута является дата окончания ремонта участка трубопровода.
* «Выполненные работы». Значением этого атрибута является информация которая содержит перечень всех выполненных работ в ходе ремонта участка трубопровода.

Описанные сущности представлены на ER-диаграмме на рис. 1.

Трубопровод

Участки трубопровода

Примечания

Ремонт на участков

Обследования с целью продления срока службы

Ревизия участков

Транспортируемые продукты

Диагностика участков

Отказы участков

Категория трубопроводов

Испытания участков

Тип изоляции

Тип и вид трубопроводов

Предприятие

Тип попутного обогрева

Тип ингибитора

Строительно-монтажные организации

Рис. 1. ER – диаграмма.

Переход от ER-диаграммы к схеме базы данных осуществляется по следующим правилам:

* сущности преобразуются в таблицы,
* связи один ко многим преобразуются во внешние ключи,
* связи многие ко многим преобразуются в таблицы.

Полученная по этим правилам схема базы данных расположена на рисунке 2.

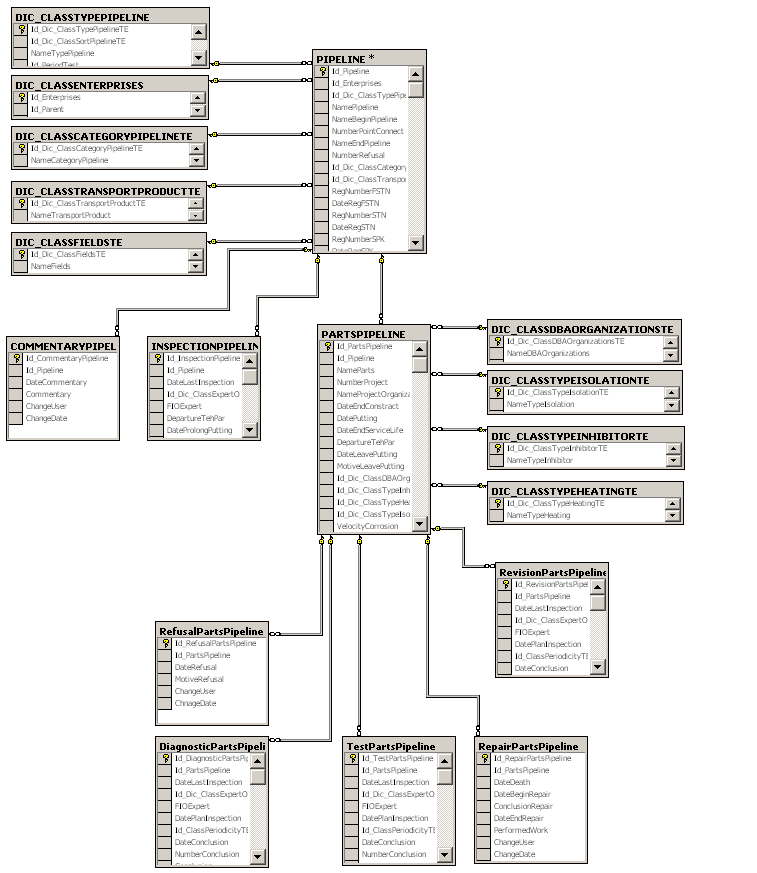


Рис. 2.

## Хранимые процедуры

Хранимая процедура для удаления записи из таблицы «Трубопроводы»:

CREATE PROCEDURE spDelPipeline(@Id uniqueidentifier) AS

DELETE FROM CommentaryPipeline

WHERE Id\_Pipeline = @id

DELETE FROM InspectionPipeline

WHERE Id\_Pipeline = @id

Delete from RevisionPartsPipeline where

RevisionPartsPipeline.id\_RevisionPartsPipeline in

(select RevisionPartsPipeline.id\_RevisionPartsPipeline from

((Pipeline inner join PartsPipeline on Pipeline.id\_Pipeline = PartsPipeline.id\_Pipeline) inner join

RevisionPartsPipeline on PartsPipeline.id\_PartsPipeline = RevisionPartsPipeline.id\_PartsPipeline)

where Pipeline.id\_Pipeline=@id)

Delete from TestPartsPipeline where

TestPartsPipeline.id\_TestPartsPipeline in

(select TestPartsPipeline.id\_TestPartsPipeline from

((Pipeline inner join PartsPipeline on Pipeline.id\_Pipeline = PartsPipeline.id\_Pipeline) inner join

TestPartsPipeline on PartsPipeline.id\_PartsPipeline = TestPartsPipeline.id\_PartsPipeline)

where Pipeline.id\_Pipeline=@id)

Delete from RefusalPartsPipeline where

RefusalPartsPipeline.id\_RefusalPartsPipeline in

(select RefusalPartsPipeline.id\_RefusalPartsPipeline from

((Pipeline inner join PartsPipeline on Pipeline.id\_Pipeline = PartsPipeline.id\_Pipeline) inner join

RefusalPartsPipeline on PartsPipeline.id\_PartsPipeline = RefusalPartsPipeline.id\_PartsPipeline)

where Pipeline.id\_Pipeline=@id)

Delete from DiagnosticPartsPipeline where

DiagnosticPartsPipeline.id\_DiagnosticPartsPipeline in

(select DiagnosticPartsPipeline.id\_DiagnosticPartsPipeline from

((Pipeline inner join PartsPipeline on Pipeline.id\_Pipeline = PartsPipeline.id\_Pipeline) inner join

DiagnosticPartsPipeline on PartsPipeline.id\_PartsPipeline = DiagnosticPartsPipeline.id\_PartsPipeline)

where Pipeline.id\_Pipeline=@id)

Delete from RepairPartsPipeline where

RepairPartsPipeline.id\_RepairPartsPipeline in

(select RepairPartsPipeline.id\_RepairPartsPipeline from

((Pipeline inner join PartsPipeline on Pipeline.id\_Pipeline = PartsPipeline.id\_Pipeline) inner join

RepairPartsPipeline on PartsPipeline.id\_PartsPipeline = RepairPartsPipeline.id\_PartsPipeline)

where Pipeline.id\_Pipeline=@id)

DELETE FROM PartsPipeline

WHERE Id\_Pipeline=@id

DELETE FROM Pipeline

WHERE Id\_Pipeline=@Id

GO

Хранимая процедура для удаления записи из таблицы «Транспортируемые продукты» :

CREATE PROCEDURE SPDelDic\_ClassTransportProductTE(@idrec uniqueidentifier) AS

DECLARE @ins\_error int, @ROWCOUNT int

BEGIN TRANSACTION TranName

DELETE FROM DIC\_CLASSTRANSPORTPRODUCTTE

WHERE Id\_Dic\_ClassTransportProductTE = @idrec

select @ROWCOUNT=@@ROWCOUNT, @ins\_error=@@error

if @ins\_error=0

begin

COMMIT TRANSACTION TranName

return (0)

end

else

begin

ROLLBACK TRAN TranName

return (1)

end

GO

## Запросы

Запрос на добавления записи в таблицу «Трубопроводы»:

SQLStr.Add('INSERT INTO dbo.PartsPipeline (Id\_PartsPipeline, Id\_Pipeline, '+

'NameParts, NameProjectOrganizations, DateEndConstract, DatePutting, '+ 'DateEndServiceLife, DateLeavePutting, DepartureTehPar, '+ 'Id\_Dic\_ClassDBAOrganizationsTE, 'Id\_Dic\_ClassTypeIsolationTE, '+ Id\_Dic\_ClassTypeHeatingTE, Id\_Dic\_ClassTypeInhibitorTE, '+

'IncutDate, Length, StellGrade, OuterDiameter, WorkingPressure, WorkingTemperature, '+

'NominalThickness, ActualThickness, VelocityCorrosion, '+

'MotiveLeavePutting, NumberProject) '+

'VALUES ('''+ Id\_PartsPipeLine +''', '+

''''+ Id\_Pipeline +''', '+

''''+ memNameParts.text +''', '+

''''+ memNameProjectOrg.text +''','+

''+ DateTimeToStrDB(deDateEndConstract.Date,' : ') +', '+

''+ DateTimeToStrDB(deDatePutting.Date,' : ') +', '+

''+ DateTimeToStrDB(deDateEndServiceLife.Date,' : ') +', '+

''+ DateTimeToStrDB(deDateLeavePutting.Date,' : ') +', '+

''''+ memDepartureTehPar.text +''', '+

Id\_ClassDBAOrganizations +', '+

Id\_ClassTypeIsolation +', '+

Id\_ClassTypeHeating +', '+

Id\_ClassTypeInhibitor +', '+

''+ DateTimeToStrDB(deIncutDate.Date,' : ') +', '+

''''+ edtLength.text +''', '+

''''+ edtStellGrade.text +''', '+

''''+ edtOuterDiameter.text +''', '+

''''+ edtWorkingPressure.text +''', '+

''''+ edtWorkingTemperature.text +''', '+

''''+ edtNominalThickness.text +''', '+

''''+ edtActualThickness.text +''', '+

''''+ edtVelocityCorrosion.text +''', '+

''''+ memMotiveLeavePutting.text +''', '+

''''+ edtNumberProject.text +''')');

Запрос на обновление записи примечания к трубопроводу в таблице «Примечания»:

MemberCommentary.Add('UPDATE dbo.CommentaryPipeline SET '+

'Commentary = '''+ Data.Commentary +''', '+

'DateCommentary = '+ ConvertFormatDate(Data.DateCommentary) +' '+

' WHERE Id\_CommentaryPipeline = '''+ Data.Id\_Commentary +''' ');

## Выбор модели базы данных

Существую три дореляционных подхода к организации баз данных:

* системы, основанные на инвертированных списках,
* иерархические системы,
* сетевые системы.

Одними из современных подходов к организации баз данных являются объектно-ориентированный и реляционный.

Данные в рамках рассматриваемой предметной области сильно структурированы. В данной работе использована реляционная модель, поскольку она является наиболее эффективной для представления такого рода данных.

## Организация взаимодействия с пользователем

Пользовательский интерфейс должен предоставлять следующие функции:

* поиск по типу;
* поиск по виду;
* поиск по наименованию трубопровода;
* поиск по категории трубопровода;
* поиск по транспортируемым продуктам;
* поиск по регистрационному номеру ФСТН;
* поиск по дате регистрации в ФСТН;
* поиск по регистрационному номеру СТН;
* поиск по дате регистрации в СТН;
* поиск по месторождению;

**3. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ**

## 

## 3.1 Выбор средства разработки

Для реализации данного проекта был выбран язык Delphi. Будучи языком высокого уровня, он всё же предоставляет программисту полный контроль над машиной, позволяет переходить на язык более низкого уровня (ассемблер). Delphi является стандартом для приложений, где нужно быстродействие и малый размер кода при достаточно глобальных масштабах проекта.

В качестве рабочей среды разработки пользовательского приложения программы использовался Delphi 2005. Выбор именно этой среды разработки был обусловлен целым рядом факторов – эта среда позволяет быстро создавать приложения различной сложности, также она поддерживает все современные наработки как в собственно языке Pascal, так и разнообразных библиотек, необходимых для разработки.

## 3.2 Выбор СУБД

Для решения поставленной задачи СУБД должна отвечать следующим требованиям:

* реляционная модель представления данных,
* поддержка многопользовательского режима работы,
* работа на платформе Windows 2000 и выше.

Предъявленным требованиям отвечают следующие СУБД:

* Microsoft SQL Server 2000,
* Oracle,
* IBM DB-2.

СУБД DB-2 может обслуживать до 64 000, а Oracle до 10 000 одновременно работающих пользователей [8]. Использование их в рамках данного проекта является не целесообразным расходованием ресурсов.

На этапе разработки системы была построена реализация с использованием базы данных на MS Access. Тестовая эксплуатация в сети Интернет показала правильность выбранных методов решения поставленной задачи. Было выявлено, что MS Access не выдерживает необходимые нагрузки, это выражается в увеличении времени ответа системы. Система интерактивная и подобные задержки недопустимы.

Из выше перечисленного следует, что в данном проекте необходимо использовать СУБД MS SQL Server 2000.

## 

## 3.3 Описание системы

Для настройки работы программы формируется файл DataBase.ini. В этом файле необходимо прописать провайдера, имя базы данных и имя сервера к которому будет происходить обращение. Также необходимо указать путь к справочникам, к которым происходит обращение при работе пользовательского приложения. Пример файла конфигурации представлен в приложении 1.

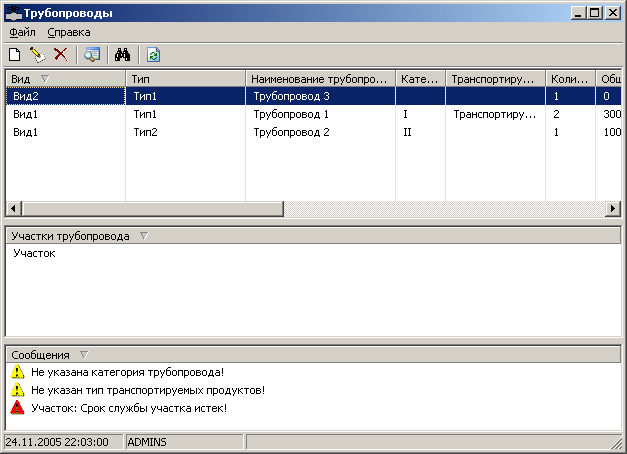
Пользовательский интерфейс

Пользовательский интерфейс системы состоит из трех панелей. В верхней части экрана расположена «Инструментальная» панель. Панель содержит кнопки для управления записями в базе данных.

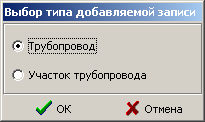
На верхней панели расположена форма «Трубопроводы». Форма предназначена для отображения основных сведений о трубопроводе.

На нижней панели расположена форма «Участки трубопровода». В процессе работы с системой отображает все участки выделенного трубопровода.

В строке состояния отображается информация, когда и кем были произведены последние изменения выделенной записи.



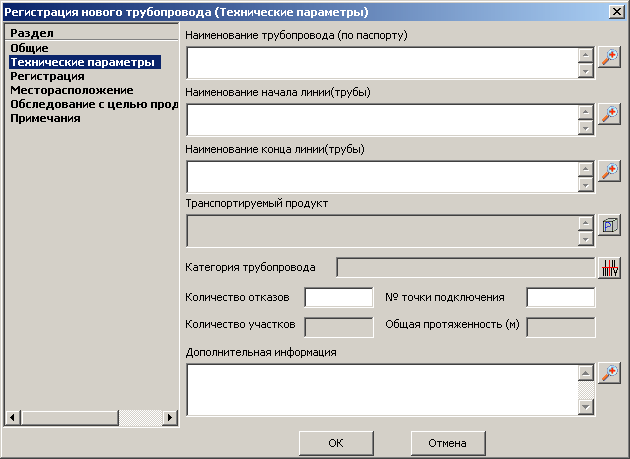
При добавление нового трубопровода или участка трубопровода, необходимо выбрать тип добавляемой записи.



Форма добавления нового «Трубопровода».

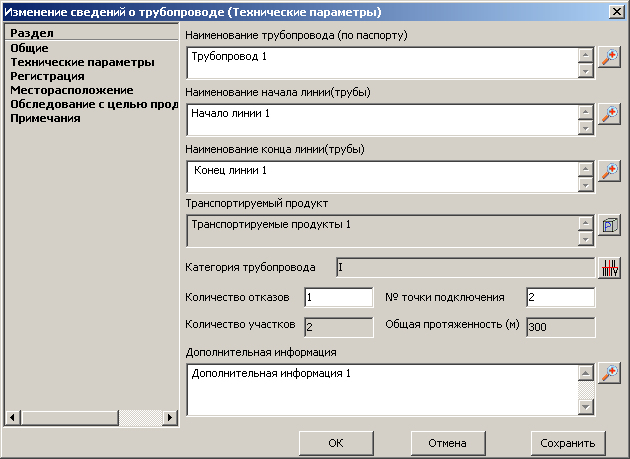
На левой панели находятся название тематических разделов данных. При выборе пользователем нужного раздела данных. На правой панели обновляется информации, соответствующая данному разделу. Поля которые однозначно определяются значениями, которые выбирается пользователем из справочника, не доступны для редактирования. Поля в которых предусмотрен ввод данных пользователем не обрабатываются на правильность ввода и имеют дополнительный редактор для удобства ввода и просмотра информации.

Внизу находятся две кнопки управления: ОК, Отмена. При нажатии кнопки Отмена, пользователь отказывается от добавления нового трубопровода. При нажатии ОК, пользователь подтверждает добавление нового трубопровода и правильность введенных данных.



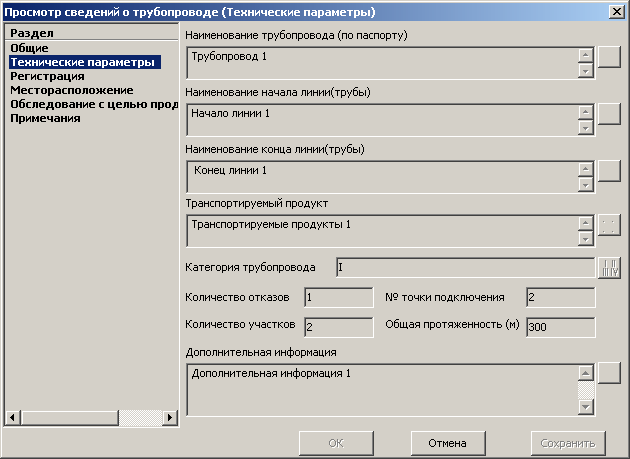
Форма редактирования «Трубопровода».

При выборе пользователем режима редактирования, поля на форме автоматически заполняются данными, которые содержаться в базе. Пользователь в этом режиме может осуществлять редактирование данных. При выборе кнопки ОК, пользователь подтверждает сохранение измененных данных. После обновления данных форма редактирования закрывается. При нажатии кнопки Отмена, пользователь завершает редактирование данных о трубопроводе без сохранения данных. При нажатии кнопки Сохранить, пользователь подтверждает сохранение данных с дальнейшим редактированием их.



Форма просмотра «Трубопровода».

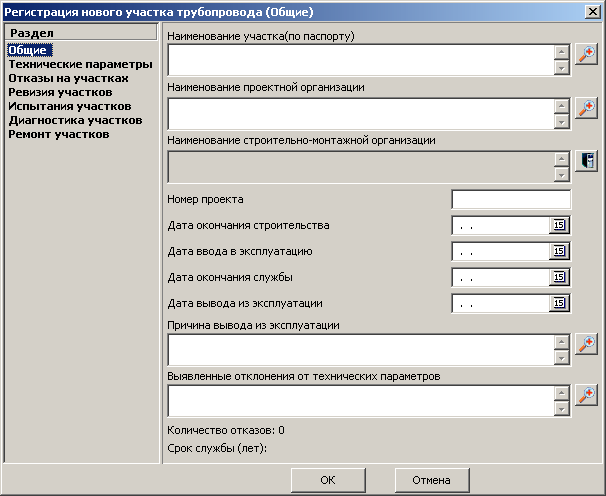
При выборе пользователем режима просмотра, поля на форме автоматически заполняются данными, которые содержаться в базе. Пользователь может осуществлять только просмотр данных в этом режиме.



Форма добавления нового «Участка трубопровода».

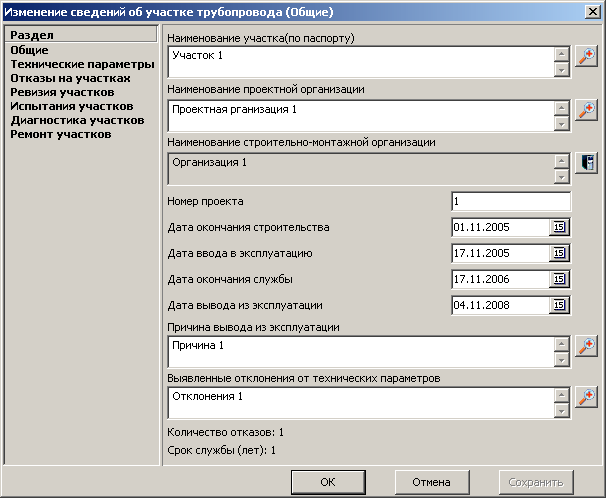
На левой панели находятся название тематических разделов данных. При выборе пользователем нужного раздела данных. На правой панели обновляется информации, соответствующая данному разделу. Поля которые однозначно определяются значениями, которые выбирается пользователем из справочника, не доступны для редактирования. Поля в которых предусмотрен ввод данных пользователем не обрабатываются на правильность ввода и имеют дополнительный редактор для удобства ввода и просмотра информации.

Внизу находятся две кнопки управления: ОК, Отмена. При нажатии кнопки Отмена, пользователь отказывается от добавления нового трубопровода. При нажатии ОК, пользователь подтверждает добавление нового трубопровода и правильность введенных данных.



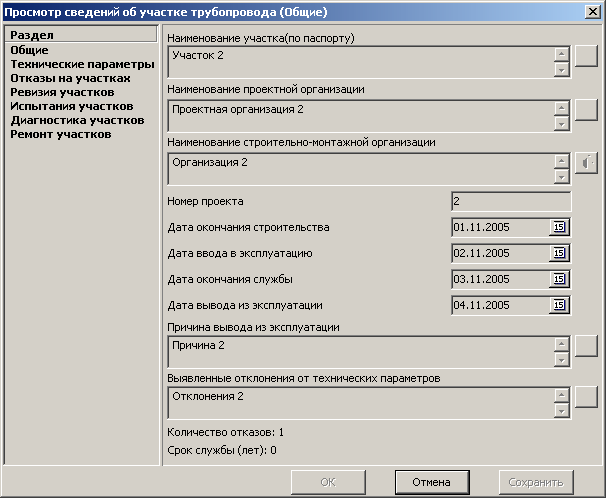
Форма редактирования «Участка трубопровода».

При выборе пользователем режима редактирования, поля на форме автоматически заполняются данными, которые содержаться в базе. Пользователь в этом режиме может осуществлять редактирование данных. При выборе кнопки ОК, пользователь подтверждает сохранение измененных данных. После обновления данных форма редактирования закрывается. При нажатии кнопки Отмена, пользователь завершает редактирование данных о трубопроводе без сохранения данных. При нажатии кнопки Сохранить, пользователь подтверждает сохранение данных с дальнейшим редактированием их.



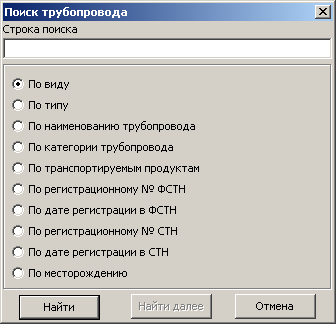
Форма просмотра «Трубопровода».

При выборе пользователем режима просмотра, поля на форме автоматически заполняются данными, которые содержаться в базе. Пользователь может осуществлять только просмотр данных в этом режиме.



Форма поиска «Трубопровода».

При поиске трубопровода необходимо выбрать по какому полю будет осуществляться поиск трубопровода. Также необходимо ввести строку поиска. Поиск происходит до первого вхождения строки поиска в указанное поле трубопровода, после чего найденная запись выделается. Для продолжения поиска необходимо нажать кнопку «Найти далее», если достигнуть конец списка, то пользователю выдается информационное сообщение о достижении конца списка. Если поиск не дал результатов пользователю выдается сообщение о том что поиск не принес результатов.



## 3.4 Синхронизация при работе в сети

При попытке редактировать или удалить запись идет проверка не редактируется ли это запись другим пользователем, если запись не находится в режиме редактирования, то она доступна только для просмотра.

## 3.5 Технические характеристики программы

Общепринятым считается указывать две конфигурации компьютера, которые используются в качестве рабочей платформы для программного продукта. Это минимальная, в которой работа с программой будет очень затруднена, и обеспечивающая лишь запуск и минимальное функционирование программы, и рекомендованная, которая позволит получить наибольшую отдачу от программы.

Минимальные требования:

* микропроцессор пятого или шестого поколения (производителей AMD, Intel)
* 16 MB ОЗУ
* 20 Мб на диске
* остальное используемое оборудование должно удовлетворять требованиям, накладываемым выбранной ОС семейства Microsoft Windows (2000 и выше)

Рекомендуемые требования:

* микропроцессор шестого поколения и выше (производителей AMD, Intel)
* 64 MB ОЗУ
* 50 Мб на диске
* остальное используемое оборудование должно удовлетворять требованиям, накладываемым выбранной ОС семейства Microsoft Windows (2000 и выше)

# ЗАКЛЮЧЕНИЕ

В курсовом проекте была поставлена задача разработки системы контроля и учета технического состояния магистральных трубопроводов транспортирующих огнеопасные вещества.

В ходе выполнения курсового проекта был проведен анализ прикладной области. В результате проведенного анализа были выявлены требования к разрабатываемой системе. На основе требований разработана база данных содержащая в себе технические характеристики трубопроводов. Система аккумулирует информацию, о трубопроводах занесенную в базу данных. Для хранения данных используется реляционная СУБД MS SQL Server 2000. В системе разработан пользовательский интерфейс, предоставляющий средства для удаленной работы с через локальную вычислительную сеть.

Возможно расширение созданного программного средства, включив в него возможность автоматического создания отчетов и актов.

# СПИСОК ЛИТЕРАТУРЫ

1. Методическое пособие по выполнению курсового проекта по курсу “Базы данных”, Просуков
2. Дейт К. Дж. Введение в системы баз данных, 7-е изд. - М.: Издательский дом "Вильямс", 2001.
3. Проектирование и реализация баз данных Microsoft SQL Server 2000. Учебный курсMCAD/MCSE, MCDBA .- Пер. с англ.-2-е изд., испр.- М.: Издательско-торговый дом «Русская редакция», 2003.-512 стр.: ил.
4. Программирование в SQL-Server 2000. Ребекка М. Риордан, ЭКОМ, Москва, 2002.
5. Microsoft MSDN.
6. Информационные и учебные ресурсы Интернет.

# Приложение 1.

**Файл конфигурации Database.ini.**

[Database]

Provider=SQLOLEDB.1

Database=PIPELINE

ServerName=KONTORA

[DLL]

DLLPath=C:\БД\bin